Predictor-based stabilization of discrete time-varying input-delay systems

A. Gonzalez A. Sala P. Albertos

*Dept. of Systems Engineering & Control, Technical University of Valencia, E-46022 Valencia, SPAIN, e-mail: angonsor@upnet.upv.es.

Abstract

This paper adapts some literature results on stabilizing discrete time-varying input delay systems to the case of having a predictor-based controller. The objective of the paper is showing that the incorporation of predictors to the referred results is relatively simple and, by means of such predictors, robustness against delay mismatch in the input channel may be improved. In this way, larger delay variation margins are proved with predictors when compared to standard memoryless state feedback, as intuitively expected, due to the extra past information available.

Key words: Unstable time-delay systems; Time-varying delay-dependent stability; h-step ahead predictor scheme; Linear matrix inequality (LMI).

1 Introduction

In control design of discrete delay systems there exist proposals in literature using memoryless state feedback [5] as well as predictor-based controllers (adapting FSA [11] to the discrete case). Predictors are, actually, dynamic systems; other dynamic compensator/observer proposals appear in [7]. In literature, stability is proved either with Lyapunov-Krasovskii functionals [5, 13] or input-output small-gain approaches [9].

In predictor-based stabilization cases, discrete time-varying delays have been approached only for stability analysis [10, 3]. However, to the authors’ knowledge, controller synthesis in the predictor-based approach has been only developed for constant and known delay by considering a related delay-free system.

The objective of this paper is showing that it is possible to adapt literature results on memoryless state feedback to the predictor case with non-constant delays. For instance, the results in [6] have been used as the basic building step of the predictor-based result. Nevertheless, it may be possible to carry out similar derivations with other literature results, such as the ones above.

This work proposes a CCL [1] algorithm for discrete predictor-based controller synthesis under time-varying input delay. The proposal reduces to convex LMI conditions for stability analysis and, also, in synthesis if some (quite conservative) changes of variable are carried out.

Numerical examples show that predictors can improve the proved delay variation interval in some cases, with respect to a static state feedback (as intuitively expected, because they use more past information than the non-predictor controllers).

2 Preliminaries

Consider the following discrete input-delay system:

\[ x_{k+1} = Ax_k + Bu_{k-d_k} \]  \hspace{1cm} (1)

where \( x_0 = \psi, \ u_k = \phi_u(\kappa), \ -d_M \leq \kappa < 0 \)

with \( \psi \) an initial condition for state and \( \phi_u(\kappa) \) represent initial conditions for the control action input \( u_k \), and \( A \in \mathbb{R}^{n \times n}, B \in \mathbb{R}^{n \times m} \) are the plant parameter matrices. The delay \( d_k > 0 \) is assumed not measurable, but known to vary, randomly, in an interval \( d_m \leq d_k \leq d_M \). The lower and upper delay bounds (\( d_m \) and \( d_M \), respectively), are assumed known. To simplify further developments, \( \phi_u(k) = 0 \) will be assumed on the sequel.

Static memoryless state feedback. If a static state-feedback control law, \( u_k = Kx_k \), is proposed, the loop equations are:

\[ x_{k+1} = Ax_k + BKx_{k-d_k} \]  \hspace{1cm} (2)
Stability analysis and controller synthesis for the above system have been explored with discrete Lyapunov-Krasovskii functionals [12]. There is an inherent conservativeness because: (a) $x$ can no longer be regarded as the “state” vector as past input information is needed for future predictions, (b) The Lyapunov-Krasovskii functionals do not usually represent with full generality the class of Lyapunov functions possibly needed [8].

**Prediction-based control.** As an alternative, consider the state-feedback prediction-based control law

$$u_k = K\hat{x}_{k+h}$$  \hspace{1cm} (3)

$$\hat{x}_{k+h} = A^h\hat{x}_k + \sum_{i=0}^{h-1} A^{h-i-1}B u_{k-i}$$  \hspace{1cm} (4)

where $\hat{x}_{k+h}$ is the $h$-step ahead state prediction with $h$, being a user-defined parameter [4,10].

### 3 Main Result

**Lemma 1** The closed-loop realization of model (1) with the control law (3) can be expressed as

$$z_{k+1} = (A + BK)z_k + A^hBKz_{k-d} - A^hBKz_{k-h}$$  \hspace{1cm} (5)

where $z_k$ is the prediction of $x_{k+h}$, i.e., $z_k = \hat{x}_{k+h}$.

**Proof:** Define the new state $z_k = \hat{x}_{k+h}$ from (4). Then,

$$z_{k+1} = A^h\hat{x}_{k+1} + \sum_{j=1}^{h} A^{h-j}Bu_{k-j}$$

Substituting $x_{k+1}$ from the system model (1) and adding and subtracting the terms $j = 0$ and $j = h$ in the right-hand-side summation, we obtain

$$z_{k+1} = A^h[Ax_k + Bu_{k-d}]$$

$$+ A\sum_{j=0}^{h-1} A^{h-j-1}Bu_{k-h+j} + Bu_k - A^hBu_{k-h}$$  \hspace{1cm} (6)

which can then be expressed as:

$$z_{k+1} = A\tilde{x}_k + Bu_k - A^hBu_{k-h} + A^hBu_{k-d}$$  \hspace{1cm} (7)

Finally, taking into account the control law $u_k = K\tilde{x}_k$, the proof is completed. \hfill \square

Note that, for constant known delays $d_k = h$, a delay-free closed loop is recovered thanks to the predictor.

**Theorem 1** Define $\tau = d_M - d_m$. The system (5) is globally stable if there exist matrices $P$, $L$, $Q$, $Q_m$, $Q_H$, $Z$, $Z_2$, $Z_1$, $M > 0$ and matrices $S_1$, $S_2$, $T_1$, $T_2$, $K$ that satisfy:

$$\begin{pmatrix} \Gamma & \tau S \\ \tau S^T - Z_M \end{pmatrix} < 0$$

$$\begin{pmatrix} \Gamma & \tau T \\ \tau T^T - Z_M \end{pmatrix} < 0$$

where

$$S = \begin{pmatrix} 0 & S_1^T & 0 & 0 & 0 \\ S_2^T & 0 & 0 & 0 & 0 \end{pmatrix}^T$$

$$T = \begin{pmatrix} 0 & T_1^T & 0 & T_2^T & 0 & 0 \end{pmatrix}^T$$

$$\Gamma = \begin{pmatrix} \Gamma_1 & 0 & \cdots & 0 & \cdots & \cdots & 0 & \cdots & 0 \\ 0 & \Gamma_2 & \cdots & 0 & \cdots & \cdots & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 0 & \cdots & \cdots & 0 & \cdots & \Gamma_m \end{pmatrix}$$

$$\Gamma_1 = -P + (\tau + 1)Q + Q_M + Q_m + Q_H - Z_2$$

$$\Gamma_2 = \begin{cases} 0 & h \leq d_m \\ Z_2 & h > d_m \end{cases}$$

$$\Gamma_4 = A^T_1, \quad \Gamma_5 = (A_1 - I)^T, \quad \Gamma_6 = -Q + T_1 + T_1^T - S_1 - S_1^T$$

$$\Gamma_7 = S_1 - S_2^T, \quad \Gamma_8 = -T_1 + T_2^T, \quad \Gamma_9 = B_1^T$$

$$\Gamma_{10} = B_1^T, \quad \Gamma_{11} = -Q_m + S_2 + S_2^T - Z_1 + \begin{pmatrix} 0 & h \leq d_m \\ -Z_2 & h > d_m \end{pmatrix}$$

$$\Gamma_{13} = -Q_h - Z_1 + \begin{cases} -Z_2 & h \leq d_m \\ 0 & h > d_m \end{cases}$$

$$A_1 = A + BK, \quad B_1 = A^hBK$$

$$Z_M = Z - \gamma^2Z_2 - (d_m - h)^2Z_1, \gamma = \min(d_m, h)$$

and

$$PL = I, \quad ZM = I$$

**Proof:**

Following the ideas in [3,6], a Lyapunov-Krasovskii function candidate is defined as

$$V(k) = V_1(k) + V_2(k) + V_3(k) + V_4(k) + V_5(k)$$

(10)
Theorem 1 can be used to obtain, for a given \( d_m \), an estimate of the maximum \( d_M \) leading to a feasible stabilizing controller. Such estimate can be found by the following CCL algorithm \([1,12]\):

- **Step 1**: Obtain some \( K \) that stabilizes the delay-free closed-loop realization with memoryless state feedback \( A + BK \); set \( d_M = d_m = h \) and find a feasible\(^1\) set \((P_0, L_0, Z_0, M_0, \cdots)\) in Theorem 1 for this \( K \). Set \( q = 0 \) and the iteration counter \( n_t = 1 \).

- **Step 2**: Solve the following LMI problem: Minimize \( \text{tr}(P_0 L_0 + P L_q + Z_0 M + Z M_q) \) subject to (8) in Theorem 1. Set \( P_{q+1} = P, L_{q+1} = L, Z_{q+1} = Z, M_{q+1} = M \).

- **Step 3**: With the feedback gain \( K \) in Step 2, check feasibility of Corollary 1. If feasible, set \( q = q + 1, n_t = 1, \) increase \( d_M \) by a small amount and return to Step 2. Otherwise (i.e., Corollary 1 incomplete), if \( n_t \) is greater than a specified number of iterations, then **stop**, else set \( q = q + 1, n_t = n_t + 1 \) and return to **Step 2**.

In order to compare predictor-based synthesis with memoryless designs, the following result is also needed.

**Corollary 4** To design \( K \) for memoryless state feedback, Theorem 1 in [6] can be also put as a CCL problem by applying suitable Schur complements and some

---

1 Note that CCL may be initialised to any feasible solution of (8) without the non-convex constraints (9). However, for \( d_M = d_m \), Corollary 2 ensures that a feasible solution for the overall problem exists so it is suggested as a sensible starting point.
algebraic manipulations. Initialization may be set to an arbitrary feasible solution of (8).

4 Numerical example

Let us consider the following open-loop unstable second order DT system (an inverted pendulum), from [2, Example 3], with a sampling period T=10ms:

\[
x_{k+1} = \begin{pmatrix} 1.0009 & 0.0100 \\ 0.1730 & 1.0009 \end{pmatrix} x_k + 10^{-3} \begin{pmatrix} -0.0018 \\ -1.7652 \end{pmatrix} u_{k-d_k}
\]

where the process state \( x_k = [x_{1k}, x_{2k}]^T \) is assumed accessible and \( d_m \leq d_k \leq d_M \).

In Table 1, the maximum achievable upper bound delay \( d_M \) that allows to find a stabilizing controller for some \textit{a priori} fixed values of \( d_m \), is depicted. With the proposed algorithm in Section 3.1, the improvement achieved with predictors (PBC) from Theorem 1 with respect to memoryless feedback gains (SSF) from Corollary 4 is shown: for \( d_m > 23 \) no feasible memoryless controller is found, whereas predictors will always find a feasible \( d_M \) (albeit perhaps \( d_M = d_m \), Corollary 2). For instance, with \( d_m = 60 \), the predictor-based controller with \( K = \begin{pmatrix} 98.5106 & 22.2021 \end{pmatrix} \) and \( b = 60 \) stabilizes the plant for \( d_m = 60, d_M = 61 \). For larger \( d_m \) only constant-delay cases are feasible with PBC.

Table 1 Calculation of maximum \( d_M \) given \( d_m \). SSF: Static state feedback, PBC: Predictor-based control (\( h = d_m \))

<table>
<thead>
<tr>
<th>Fixed ( d_m )</th>
<th>0</th>
<th>10</th>
<th>23</th>
<th>30</th>
<th>40</th>
<th>50</th>
<th>60</th>
</tr>
</thead>
<tbody>
<tr>
<td>( d_M ) (SSF)</td>
<td>22</td>
<td>23</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>( d_M ) (PBC)</td>
<td>22</td>
<td>25</td>
<td>31</td>
<td>36</td>
<td>44</td>
<td>52</td>
<td>61</td>
</tr>
</tbody>
</table>

In order to obtain the above results, the CCL stopping parameter was set to \( n_1 < 30 \) iterations.

5 Conclusions

This paper presents a stabilization method for discrete time-varying input delay systems with predictor-based controllers, by adapting some literature results on memoryless state feedback to the predictor case. Such memoryless controllers are a particular case, as well as the constant-delay case (stable \( A + BK \)). The controller gains need to be solved by, for instance, CCL.

As intuitively expected, the use of predictors improves the achievable delay variation range over static feedback controllers given the larger past information available in the former case.

Acknowledgements: The authors are grateful to the financial support of projects PROMETEO/2008/088 and DPI2011-27845-C02-01 from Valencian and Spanish governments, respectively.

References


Appendix: Proof of Theorem 1.

Let us first introduce an augmented state defined as \( \mu^T(k) = [z(k), z(k-d_k), z(k-d_m), z(k-d_{M}), z(k-h)] \) and \( \Theta = \)
\[ V_1(k) = \mu^T(k)\Theta^T P\Theta(k) + z^T(k)Pz(k) \]

Applying the Tchebyschev inequality (Lemma 1 in [6]), with \( \Delta V_1(k) \) defined in (10), the increments of \( V_1 \) to \( V_5 \) can be written respectively as:

\[
\begin{align*}
\Delta V_5(k) &= \mu^T(k)\Theta^T P\Theta(k) - z^T(k)Pz(k) \\
\Delta V_3(k) &= (1+\gamma)[z(k)]^T Q[z(k)] - z^T(k-d_k)Qz(k-d_k) \\
\Delta V_4(k) &= z^T(k)(Q_m+Q_M+Q_h)z(k) - z^T(k-d_m)Q_Mz(k-d_m) - z^T(k-h)Q_hz(k-h) \\
\Delta V_2(k) &= \nu(k)^T(\gamma^2 Z_2 + (d_m-h)^2 Z_1 + \gamma Z_M)\nu(k)
\end{align*}
\]

\[
\begin{align*}
\Delta V_1(k) &= \gamma \sum_{i=k}^{k-d_m-1} \nu(i)^T Z_1 \nu(i) - |d_m-h| \sum_{i=k-d_m}^{k-d_m} \nu(i)^T Z_1 \nu(i) - \sum_{i=k}^{k-d_m-1} \nu(i)^T Z_M \nu(i)
\end{align*}
\]

Applying the Tchebyschev inequality (Lemma 1 in [6]), with the convention that \( \sum_{i=a}^{b} f(i) = 0 \) if \( b < a \), it yields

\[
\begin{align*}
-\gamma \sum_{i=k}^{k-d_m-1} \nu(i)^T Z_1 \nu(i) &\leq - \sum_{i=k}^{k-d_m-1} \nu(i)^T Z_2 \nu(i) \\
-|d_m-h| \sum_{i=k-d_m}^{k-d_m} \nu(i)^T Z_1 \nu(i) &\leq - \sum_{i=k-d_m}^{k-d_m} \nu(i)^T Z_1 \nu(i) \\
-\sum_{i=k-d_m}^{k-d_m} \nu(i)^T Z_M \nu(i) &\leq - \sum_{i=k-d_m}^{k-d_m} \nu(i)^T Z_M \nu(i)
\end{align*}
\]

For some matrices defined as \( S_0 = \begin{pmatrix} 0 & S_T^T & S_q^T \\ 0 & T_T^T & 0 \end{pmatrix} \), with appropriate dimensions, it yields

\[
0 = 2\mu^T(k)S_0[z(k-d_m) - z(k-d_k)] - \sum_{i=k-d_k}^{k-d_m-1} \nu(i) \leq 2\mu^T(k)S_0[z(k-d_m) - z(k-d_k)] + \sum_{i=k=d_m}^{k-d_m-1} \nu(i) \]

\[
0 = 2\mu^T(k)T_0[z(k-d_k) - z(k-d_M)] - \sum_{i=k-d_M}^{k-d_m-1} \nu(i) \leq 2\mu^T(k)T_0[z(k-d_k) - z(k-d_M)] + \sum_{i=k-d_M}^{k-d_m-1} \nu(i)
\]

Note that if \( d_k = d_m \) or \( d_k = d_M \) the above expressions reduce to the tautology \( 0 \leq 0 \) respectively.

Including the last equivalences, the forward difference \( \Delta V(k) \) can be put as

\[
\begin{align*}
\Delta V(k) &\leq \mu^T(k)\Theta^T P\Theta(k) + (d_k-d_m)\mu^T(k)S_0Z_M^{-1}S_T^T\mu(k) \\
&\quad + (d_M-d_k)\mu^T(k)T_0Z_M^{-1}T_T^T\mu(k) \leq 0
\end{align*}
\]

Now, \( \Omega \) may be expressed as \( \Omega = \bar{\Omega} + \Theta^T \Phi^T Z \Phi \) by a suitable choice of \( \bar{\Omega}, \Theta, \Phi \) (details omitted for brevity), so carrying out the two Schur complements, give \( \Gamma \) in Theorem 1, and the following is obtained:

\[
\Gamma + (d_k-d_m)S_0Z_M^{-1}S_T^T + (d_M-d_k)T_0Z_M^{-1}T_T^T < 0
\]

which is affine in \( d_k \) so it will hold if it does for \( d_k = d_m \) and \( d_k = d_M \) [6], so it yields the LMI constraints: \( \Gamma + (d_k-d_m)S_0Z_M^{-1}S_T^T < 0 \) and \( \Gamma + (d_M-d_k)T_0Z_M^{-1}T_T^T < 0 \). Applying Schur complement, the final result (8) is obtained\(^2\).

\[^2\] Note that, actually, \( S \) and \( T \) might be replaced by more general \( S(d_k) \) and \( T(d_k) \) obtaining a less conservative version of Theorem 1 by evaluating (12) for each possible value of \( d_k \). Anyway, as the approach is computationally expensive and not directly in the scope of this work, it will not be pursued any further.